



НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
УНИВЕРСИТЕТ

# Системный дизайн современных приложений

Лекция №2  
Высокоуровневый дизайн, принципы System  
Design и архитектуры

# Сбор и анализ требований

## Цели

Понять задачу, вывести заказчика на чистую воду, уточнить масштабы решения.

Требование — это представление потребности.

## Важные точки

1. Отсев
2. Scope Refinement
3. Функциональные требования
4. Нефункциональные требования
5. Ограничения
6. Расчет нагрузки





# Расчет нагрузки

---

## СКОЛЬКО АПТЕК В МОСКВЕ?

### Важно для интервью

1. Пользовательский трафик, сценарии использования, количество RPS
2. Сетевой трафик и соединения
3. Нагрузка на вычислительную мощность
4. Необходимое дисковое пространство для хранения данных и расчет потенциального прироста объема хранилища

### Система IoT с 50 датчиками

Параметры датчиков:

- Каждый датчик отправляет 10 МВ данных каждый рабочий день (8 часов)
- Протокол: MQTT (легковесный, для IoT).
- Глубина хранения: горячая - 3 дня, холодная - 5 лет

RPS? Сетевая нагрузка?

Диск?



# Новая и старая система

---

На что обращаем внимание?

Новая:

1. Прогнозный рост
2. Тех. радар компании
3. Стоимость на рынке

Старая:

1. План перехода
  - а. Миграции
  - б. Паттерны изменения архитектуры



# Резюме

---

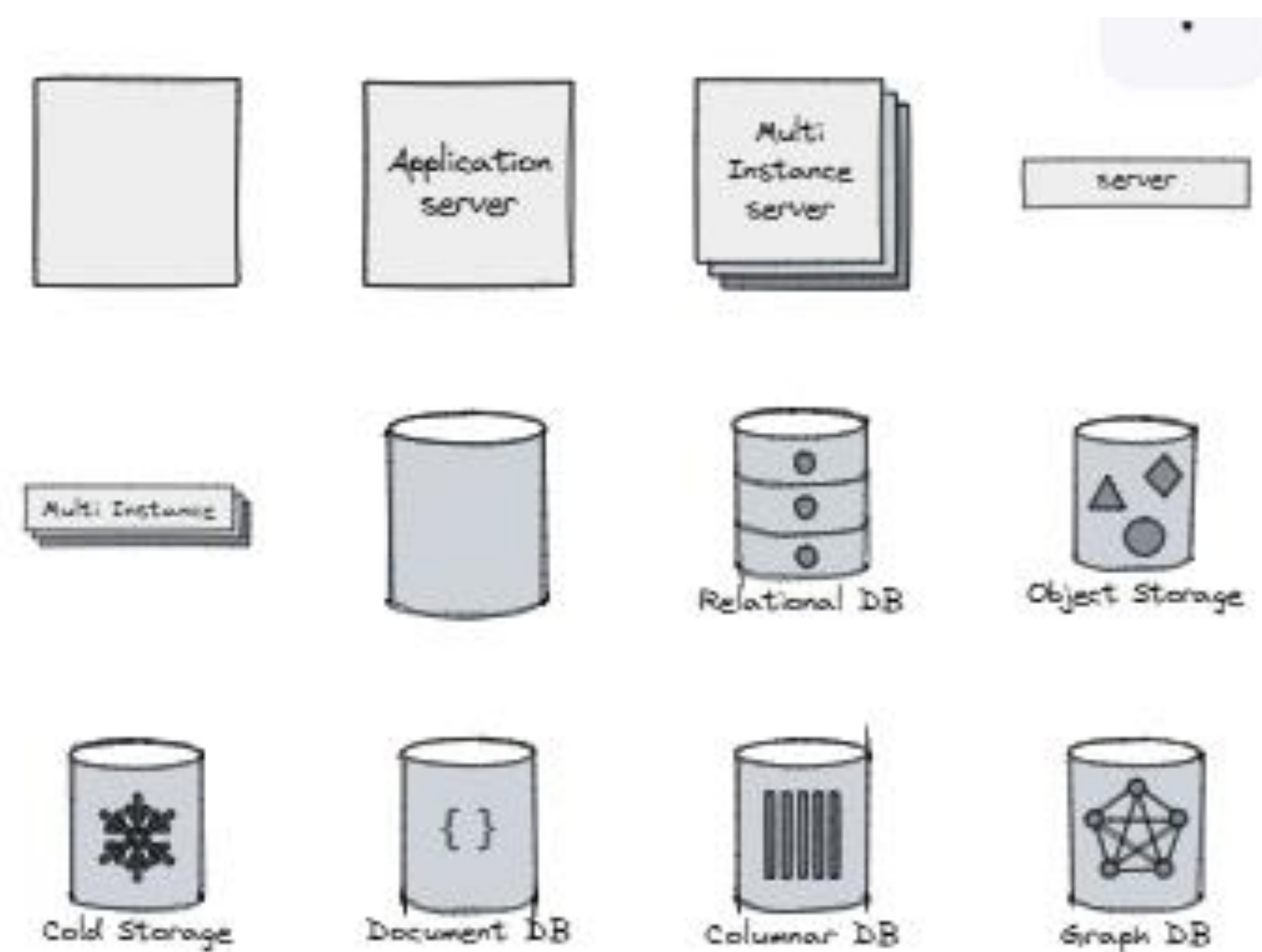
1. Не лезть сразу с техникой
2. Постоянно уточнять, рассуждать вслух
3. После определения основных сценариев обязательно уточнить corner
4. Быть адаптивным и предлагать альтернативные сценарии
5. Интервьюер - одновременно и бизнес, и член команды
6. Не переходите к рисованию пока не уточните требования



# Next Step

Собрали требования, что дальше?

квадраты притаились





# Архитектура vs System Design

---

**System Design** - это процесс проектирования архитектуры системы на уровне детализации, включающий в себя решения по распределению функциональности, определению интерфейсов и протоколов, выбору технологий и т.д.

**Архитектура ПО** - это высший уровень абстракции проектирования системы, включающий в себя определение целей, принципов, стратегий, решений и взаимодействия между компонентами системы на уровне высокой абстракции.





# Основные аспекты SD

---

1. Масштабируемость
2. Производительность
3. Надежность
  - а. Отказоустойчивость
  - б. Доступность
4. Безопасность
5. Адаптивность
6. Управляемость и мониторинг
7. Интеграции



# Масштабируемость

## Что?

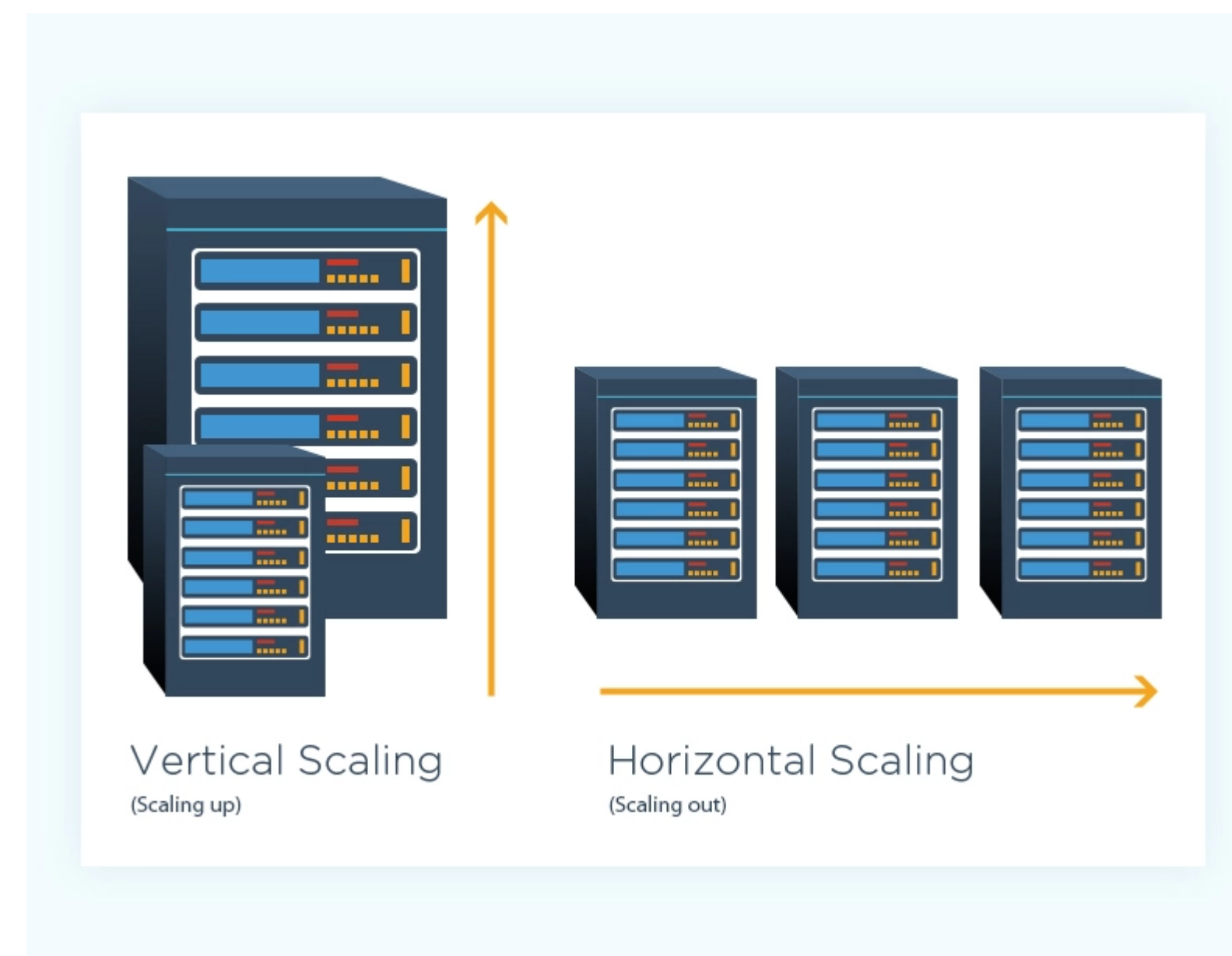
Определяет способность системы справляться с ростом какого-либо аспекта своей деятельности

## Для чего?

Удерживать тот же уровень обслуживания с ростом нагрузки

## Виды

- Горизонтальная: хрущевки
- Вертикальная: лахта-центр





# Масштабируемость

Критерий	Вертикальное	Горизонтальное
Предел роста	Ограничен железом	Теоретически неограничен
Надежность	Низкая (ботлнек)	Высокая
Стоимость	Экспоненциально растёт	Линейно растёт
Гибкость	Низкая	Высокая (автоскейлинг)
Сложность	Низкая	Высокая (балансировщики, шардирование)

# Производительность





# Надежность и доступность

Availability Percentage (Uptime)	Nines of Availability	Allowed Downtime			
		Per Year	Per Month	Per Week	Per Day
90%	One Nine	36.5 days	72 hours	16.8 hours	2.4 hours
95%	One Nine	12.85 days	1.5 days	8.4 hours	1.2 hours
99%	Two Nines	3.65 days	7.20 hours	1.68 hours	14.4 minutes
99.5%	Two Nines	1.83 days	3.60 hours	50.4 minutes	7.20 minutes
99.9%	Three Nines	8.76 hours	43.2 minutes	10.1 minutes	1.44 minutes
99.95%	Three Nines	4.38 hours	21.6 minutes	5.04 minutes	43.2 seconds
99.99%	Four Nines	52.56 minutes	4.32 minutes	60.5 seconds	8.64 seconds
99.999%	Five Nines	5.26 minutes	25.9 seconds	6.05 seconds	0.87 seconds



# Основные сущности

---

1. Клиент
2. Сервис
3. Интеграция
4. База данных



НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
УНИВЕРСИТЕТ